

SOFTWARE MAINTENANCE

TECHNICAL FIELD

The invention relates to maintaining software on a computer system and particularly to methods and systems for determining which maintenance items are already installed on a system, which items need to be added and insuring that prerequisite and corequisite items are appropriately handled.

BACKGROUND OF THE INVENTION

Computer systems of all types require periodic and sometimes unexpected or unplanned software maintenance. This applies to personal computers of all types e.g. portables and desktops as well as servers, and mainframes. When the computer system is large, expensive, or serving many users, the maintenance operations must necessarily be performed quickly and efficiently to minimize disruption of service and possible loss of revenue due to the computer system being unavailable during the planned or unplanned maintenance operation. Furthermore, a large system may have a large number of software items which were installed over a long period of time each one having different maintenance levels. Each item may also have various prerequisite items which must be installed or removed prior to installing that particular item or corequisite items which must be installed or removed during the same session or previously.

Software maintenance items may be distributed in portable media such as magnetic tapes, floppy disks, CD roms, or they may be distributed over a network or other communication channels including the internet, dial-up through a telephone service, via
5 fiber optic cable, or other data channels known in the art.

Performing software maintenance on computer systems, particularly large systems is therefore a complicated and difficult procedure, prone to errors which can disable a valuable resource for periods much larger than can reasonable be planned.
10 A number of solutions have been developed, however, none have be found to be completely satisfactory for addressing the problems described above.

For example, Tyra et al. in US Patent 5,493,682, incorporated herein by reference, describe a system for software maintenance based on object oriented programming concepts.
15 However a considerable amount of definition is needed for events required to maintain a software object including event identifiers and event conditions; object definitions including object identifier, attributes, and methods to be performed; and
20 dependency objects. Software already installed on many systems may not have such definitions so that creation of such definition may be a prohibitively burdensome task in order to use Tyra's maintenance system.

Bizuneh et al. in US Patent 5,805,891 describe a system for
25 distributing a software maintenance module by first verifying and testing the module prior to distributing it to the end users.

Glowny et al. in US Patent 5,805,897 describe a system for installing software on a remote workstation in a network having a plurality of workstations including a local workstation. Installation at the remote workstation is initiated and
5 controlled from the local workstation by communicating over the network.

Nakagawa et al. in US Patent 5,835,911 describe a software distribution and maintenance method using a plurality of user computers and a vendor computer having a software library. A
10 user computer sends an inquiry over the network to the vendor computer for the latest configuration of an object software. The inquiry may be made periodically or in response to a user initiating the software. The vendor computer receives the inquiry and generates update instruction information about the latest version in the software library. The instruction
15 information and a copy of the latest version object software are returned to the user computer via the network where the instruction information is processed to update the object software. Processing may include compiling and linking if
20 necessary.

Nishiyama et al. in US Patent 5,859,977 describe a network system having both a processing network and a separate maintenance network having high reliability and high
responsiveness. All software maintenance is carried out using
25 the maintenance network thereby providing an overall system with high reliability and expendability.

Ahmad in US Patent 6,029,258 describes trouble shooting software for a software application. A user launches the trouble shooting software when encountering a problem with the application. The trouble shooter attempts to find a problem solution in an information store of problem solutions. If located, the solution is passed to the user to install. If no solution is found the trouble shooter connects to an internet trouble shooting site and server to obtain additional solutions. If no solution is found there, the trouble shooting software then recommends that the provider of the application be contacted about the problem.

deCarmo in US Patent 5,574,905 describes a method of editing a multimedia file having segments located in different data files, through use of a linked-list structure. Some editing operations therefore result in changes to the linked-list structure rather than the file itself, avoiding making file copies.

Kiichi in Japanese Patent JP3237530 describes a system for distributing software updates. An update object list file is received by a personal computer from a host computer. The personal computer then successively receives files of all update modules based on this object list file and updates its software.

Despite the aforementioned developments there remains a need for an improved software maintenance process.

OBJECTS AND SUMMARY OF THE INVENTION

It is therefore a principal object of the present invention to enhance the software maintenance art by providing a method of maintaining software on a computer system with enhanced capabilities.

It is another object to provide a system for maintaining such software wherein enhanced capabilities are possible.

It is yet another object to provide a computer system for maintaining software which can be applied in a facile manner.

It is a further object to provide a computer program product with enhanced capability to facilitate maintaining software.

These and other objects are attained in accordance with one embodiment of the invention wherein there is provided a method of maintaining software on a computer system, comprising the steps of, bringing up first and second host sessions on a computer system, starting in the first host session, a software recording application having data on first maintenance items applied to the computer system, starting in the second host session, a database application having a database of second maintenance items including prerequisite items and corequisite items corresponding to each of the second maintenance items, activating a maintenance application on the computer system, entering a first list of third maintenance items in the maintenance application, searching the database for the prerequisite items and corequisite items

corresponding to each of the third maintenance items on the first list, and adding the corresponding prerequisite items and corequisite items to the first list, thereafter determining from the software recording application which items on the first list have already been received, and adding those items not received to an order list, and ordering, receiving, and applying the items on the order list.

In accordance with another embodiment of the invention, there is provided a system for maintaining software on a computer system, comprising, means for bringing up first and second host sessions on a computer system, a software recording application having data on first maintenance items applied to the computer system, a database application having a database of second maintenance items including prerequisite items and corequisite items corresponding to each of the second maintenance items, a maintenance application having a first list of third maintenance items, means for searching the database for the prerequisite items and corequisite items corresponding to each of the third maintenance items on the first list, and adding the corresponding prerequisite items and corequisite items to the first list, means for thereafter determining from the software recording application which items on the first list have already been received, and adding those items not received to an order list, and means for ordering, receiving, and applying the items on the order list.

BRIEF DESCRIPTION OF THE DRAWING

The Figure is a flowchart in accordance with one embodiment of the invention.

BEST MODE FOR CARRYING OUT THE INVENTION

5 For a better understanding of the present invention, together with other and further objects, advantages and capabilities thereof, reference is made to the following disclosure and the appended claims in connection with the above-described drawing.

10 In the Figure there is shown a flowchart 10 of a method for software maintenance in accordance with the present invention. In step 12 two host sessions are brought up on a computer system. The computer system is normally the computer system on which one wants to perform software maintenance, however this is not
15 required in order to practice the invention. The computer system may be a mainframe, personal computer, laptop, portable or any other type of computer system on which it is desired to perform software maintenance. The computer system may further include an operating system such as an OS/390 architecture operating system
20 and may also include associated software program products such as JESx and networking software. The computer system may also include a data terminal and a supporting software such as an IBM program called Personal Communication/3270. The computer system is capable of bringing up host sessions such as may be provided
25 by its operating system software or by some other software

installed on the computer system.

In step 14 a software recording application is brought up in a first session. The recording application has data on first maintenance items which have been applied to the computer system in the past. One example of such a recording application is a program provided by IBM Corp. known as SMP/E. This program can record what software has been put on, track such software, record what software has been taken off, and record what software has been cloned, all on an OS/390 architecture system. The software recording application includes software for determining whether any particular maintenance item has already been received by the computer system.

In step 16 a database application is started in the second session. The database application has a database of second maintenance items known as PTF's. The database also includes a listing of prerequisite and corequisite items for each second maintenance item. The database may also include descriptions known as APARS, of known problems in software items and mention of known fixes. One example of such a database application is an IBM program called Service Link. The database application includes a capability for searching the database of second maintenance items.

In step 18 a maintenance application is activated on the computer system. The maintenance application may be written in the REXX language and run in a window on the computer system. In step 20 a first list of third maintenance items are entered into

the maintenance application. This list may be a list of maintenance items needed to be put on to the computer system. The list may be entered via keyboard, importing a file, selecting from some other list, or any other method known in the art for entering a list into an application.

In step 22 the database of the database application started in step 16 is searched for prerequisite items and corequisite items corresponding to each maintenance item on the list entered in step 20. The prerequisite and corequisite items found in this searching are added to this list. Any particular item may therefore appear multiple times on the list. Such duplicates may be determined and removed to improve efficiency of the method, although this is not required by the scope of the invention.

In step 24 it is determined from the software recording application stated in step 14, which items on the list have already been received and those not received are added to an order list. Received shall be taken to mean stored in a file on the computer system. The item may have been stored in a file when the computer system was originally provided by the computer manufacturer. It may have been stored as a result of an earlier use of the present method, or as a result of some other maintenance procedure, or a file may have been provided in any other way known in the art for providing a file.

In step 26, the items on the order list are ordered, received, and applied. Applied shall mean that a received item has been installed on the computer system or updated an earlier

